

Case Study:

Download and compile PostgreSQL, and open in Eclipse, using instructions provided here. Physical Storage in PostgreSQL, Look inside above link for information on file layout, database page layout, and free space map (and lots more details src/backend/storage/freespace/README) and how PostgreSQL stores oversized attributes using the TOAST technique.

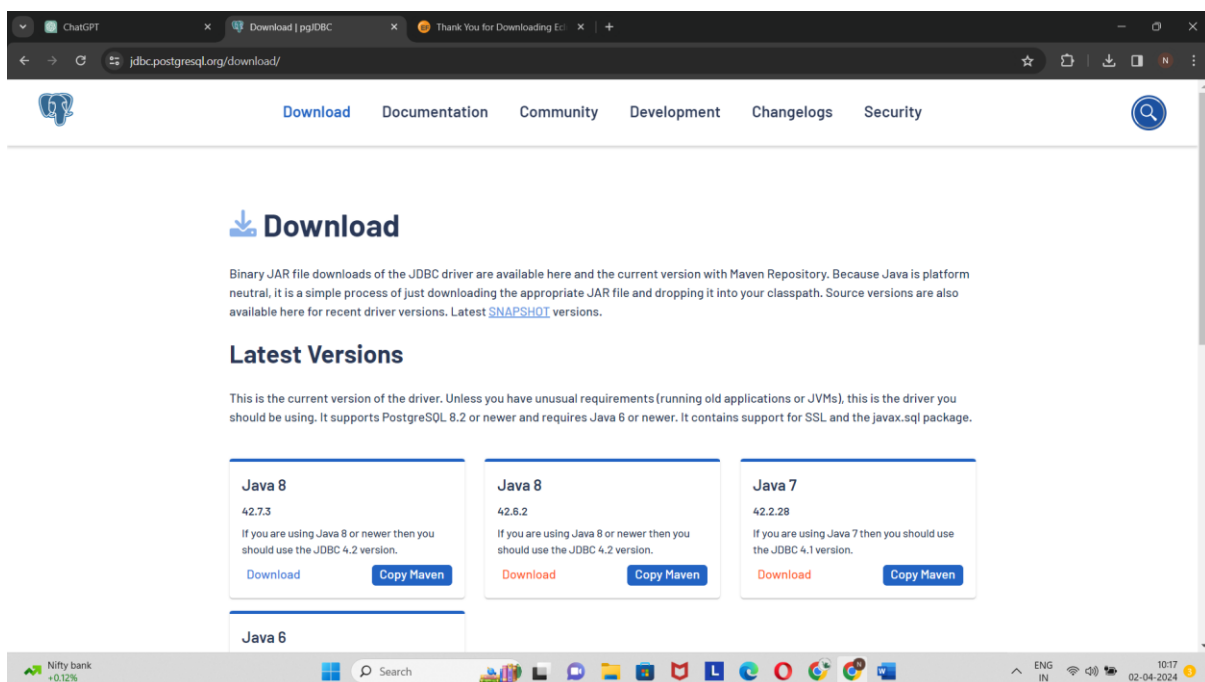
Answer:

To compile the PostgreSQL in Eclipse we have to download an external file of java that is postgresql.jar.

Then we have to add it in the project we are building and then the PostgreSQL can be compiled into the Eclipse.

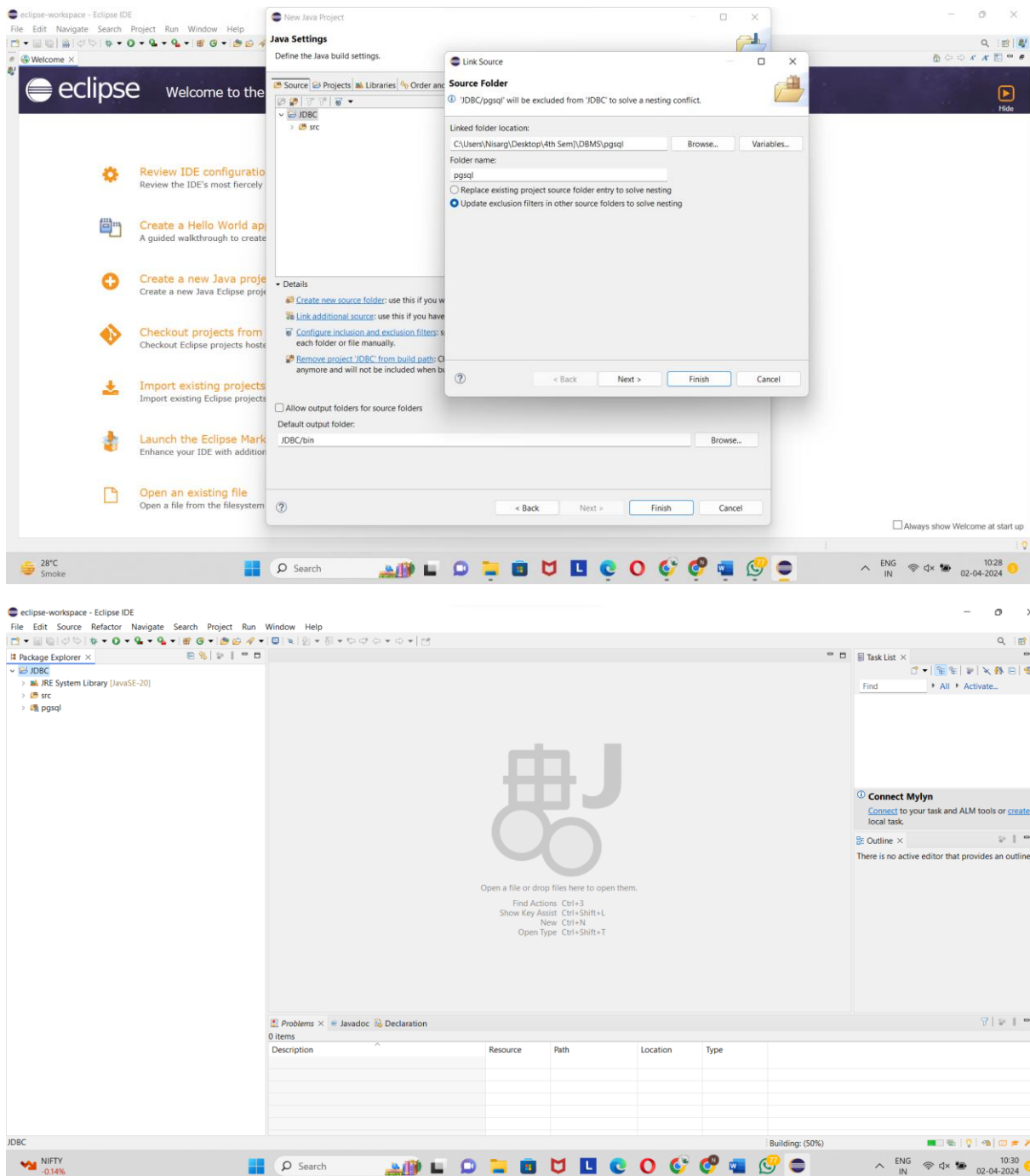
The File can be Downloaded from jdbc(Java Database Connectivity):

<https://jdbc.postgresql.org/>

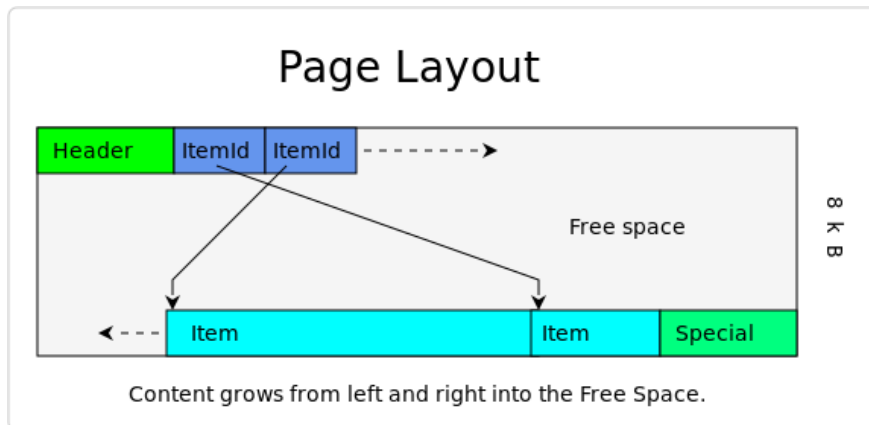


2. While Creating the Eclipse project.

The Downloaded JDBC file should be added in the dependencies of Eclipse Project.
Below Image describes the process to add the file in Eclipse



File Page Layout in PostgreSQL:



1. Tablespaces:

- Tablespaces are directories on the file system where PostgreSQL stores data files.
- PostgreSQL can have multiple tablespaces, each with its own physical location on the disk.
- By default, PostgreSQL stores all data in the "pg_default" tablespace unless otherwise specified.

2. Databases:

- Databases in PostgreSQL are logical containers for objects like tables, views, functions, etc.
- Each database can have its own set of tablespaces.
- Users can create multiple databases within a PostgreSQL instance, each isolated from the others.

3. Schemas:

- Schemas are used to organize database objects into logical groups.
- Each database can contain multiple schemas.
- Schemas are helpful for organizing and managing objects within a database, especially in complex applications with many tables and other objects.

4. Tables:

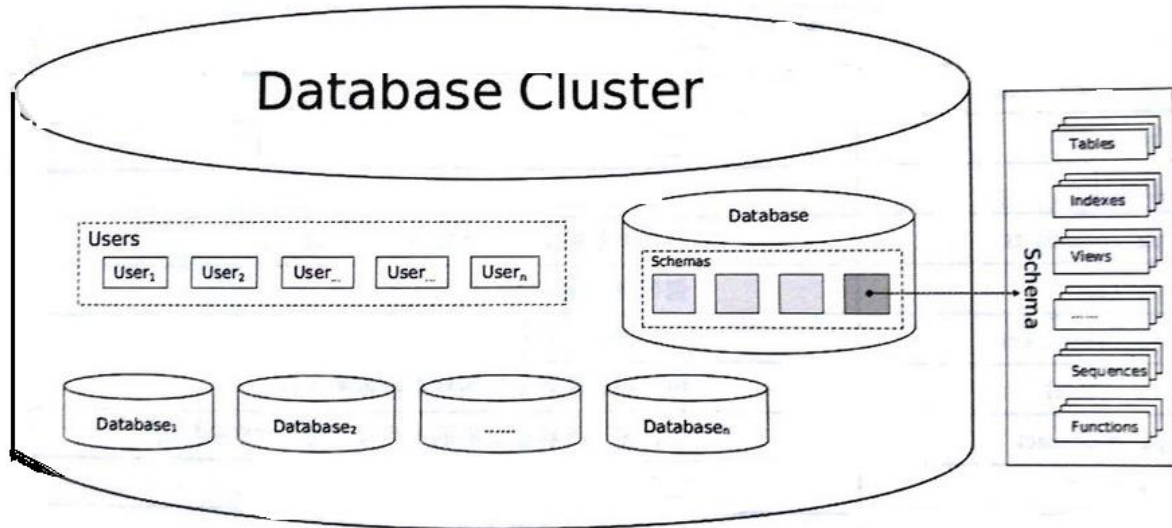
- Tables are the primary storage objects in PostgreSQL where data is stored.
- Each table consists of rows and columns, with each column having a specific data type.
- Tables can be organized into schemas and are associated with a particular namespace within a schema.

5. Indexes:

- Indexes are used to speed up data retrieval operations by providing quick access to rows based on the values of certain columns.
- PostgreSQL supports various types of indexes, including B-tree, hash, GiST, GIN, BRIN, and SP-GiST indexes.

- Indexes can be created on one or more columns of a table to optimize queries involving those columns.

Database Layout in PostgreSQL:



All the databases in PostgreSQL are stored in a Database Cluster.
And Every database contains the schema of the tables.

In PostgreSQL, data storage involves tablespaces, databases, schemas, tables, and indexes. Tablespaces are disk directories, databases contain schemas, schemas group objects, tables store data, and indexes speed up data retrieval. Physical data is organized into pages, typically 8 KB, stored in tablespaces. PostgreSQL's documentation offers detailed insights into storage layout, including data file organization, table/index structures, and storage parameters for performance and resource management.

1. Tablespaces:

- Tablespaces are directories on the file system where PostgreSQL stores data files.
- PostgreSQL can have multiple tablespaces, each with its own physical location on the disk.
- By default, PostgreSQL stores all data in the "pg_default" tablespace unless otherwise specified.

2. Databases:

- Databases in PostgreSQL are logical containers for objects like tables, views, functions, etc.
- Each database can have its own set of tablespaces.
- Users can create multiple databases within a PostgreSQL instance, each isolated from the others.

3. Schemas:

- Schemas are used to organize database objects into logical groups.

- Each database can contain multiple schemas.
- Schemas are helpful for organizing and managing objects within a database, especially in complex applications with many tables and other objects.

4. Tables:

- Tables are the primary storage objects in PostgreSQL where data is stored.
- Each table consists of rows and columns, with each column having a specific data type.
- Tables can be organized into schemas and are associated with a particular namespace within a schema.

5. Indexes:

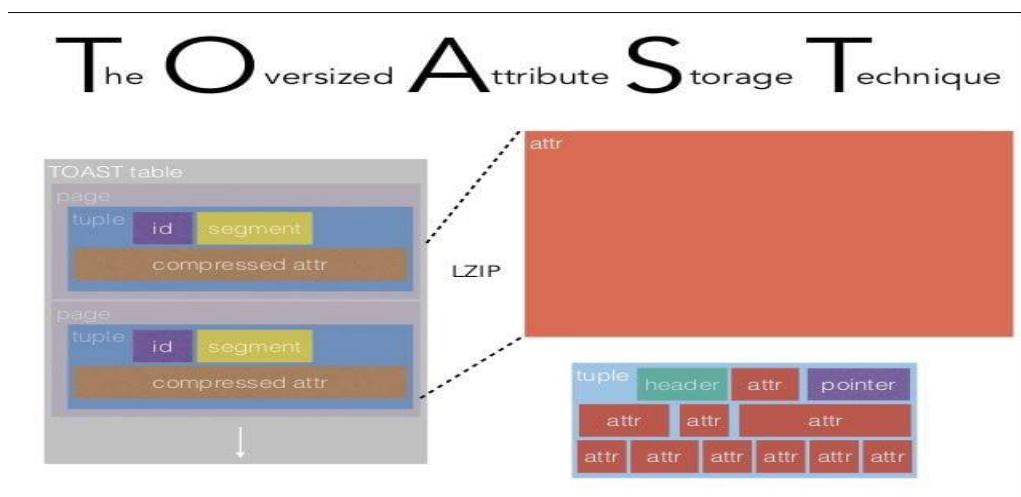
- Indexes are used to speed up data retrieval operations by providing quick access to rows based on the values of certain columns.
- PostgreSQL supports various types of indexes, including B-tree, hash, GiST, GIN, BRIN, and SP-GiST indexes.
- Indexes can be created on one or more columns of a table to optimize queries involving those columns.

FreeSpace Mapping in PostgreSQL:

Before PostgreSQL 8.4, the FSM is maintained in memory with the limitation of fixed-size. Now, all FSM files are on disk and can be extended in the same way as heap segmentation files. The FSM files will be updated during an insertion and a Vacuum process either periodically or triggered manually. In some conditions, the FSM may need to be rebuilt, for example, if a leaf node's value is less than the root node or if there is truncate operation.

FSM file is mainly used to quickly locate a page for insert operation and in the meantime it helps to speed up the insertion by trying to insert records in existing pages instead of extend a new page. By doing this, it not only help speed up the insertion but also help improve the selection performance. Moreover, the user to implement various strategies, like preferring pages closer to a given page, or spreading the load across the table. (Source: <https://www.highgo.ca/2020/10/23/free-space-mapping-file-in-details/>)

TOAST(The Oversized-attribute storage technique) Method:



TOAST (The Oversized-Attribute Storage Technique) in PostgreSQL is a mechanism designed to handle large data values that exceed a certain threshold size, typically around 2 KB. When a row in a table contains such large values, PostgreSQL automatically moves them to separate TOAST tables, while storing only small pointers, known as TOAST pointers, within the main table row.

This technique optimizes storage and access efficiency by reducing the storage overhead associated with large values. Instead of storing the entire large value inline within the table row, PostgreSQL stores a compact reference, which points to the location of the actual data in the TOAST table. The TOAST table itself stores the large values in a compressed and/or out-of-line format, ensuring efficient use of disk space.

TOAST operates transparently, meaning that users interact with the data as if it were stored inline within the table row. When large values are accessed, PostgreSQL seamlessly retrieves and reconstructs the original data from the TOAST tables, presenting it to the user as if it were never moved.

This feature is particularly beneficial for databases that handle large text or binary data, such as documents, images, or multimedia files. It helps PostgreSQL efficiently manage and optimize the storage of such data, minimizing storage overhead and improving overall database performance.

Summary:

The process of integrating PostgreSQL with Eclipse involves first installing PostgreSQL by downloading the appropriate version for your operating system, following the installation instructions, and ensuring all necessary dependencies are met. Once PostgreSQL is installed, integration with Eclipse, a Java Integrated Development Environment (IDE), entails adding JDBC (Java Database Connectivity) drivers to your Eclipse project's build path and writing Java code to establish connections with the PostgreSQL database, execute queries, and manipulate data. Meanwhile, within PostgreSQL, the TOAST (The Oversized-Attribute Storage Technique) mechanism efficiently handles large data values that surpass a predetermined threshold size by automatically relocating them to separate TOAST tables and storing compact pointers, known as TOAST pointers, within the main table row. This technique optimizes storage and access efficiency, particularly for large text or binary data, as it reduces storage overhead and ensures the seamless retrieval and reconstruction of original data from the TOAST tables when accessed. Therefore, the integration of PostgreSQL with Eclipse not only enables streamlined database management and interaction through Java applications but also leverages PostgreSQL's advanced storage techniques like TOAST to effectively manage and optimize storage of large data values.